

ANALISIS PENAMBAHAN NILAI MOMENTUM PADA PREDIKSI PRODUKTIVITAS KELAPA SAWIT MENGGUNAKAN BACKPROPAGATION

Eka Irawan¹, M. Zarlis², Erna Budhiarti Nababan³

Magister Teknik Informatika, Universitas Sumatera Utara

Jl. Universitas No.9A Kampus USU, Medan, Sumatera Utara-Indonesia

ekaatb09@gmail.com

Abstrak— Algoritma backpropagation merupakan multi layer perceptron yang banyak digunakan untuk menyelesaikan persoalan yang luas, namun algoritma backpropagation juga mempunyai keterbatasan yaitu laju konvergensi yang cukup lambat. Pada penelitian ini penulis menambahkan parameter learning rate secara adaptif pada setiap iterasi dan koefisien momentum untuk menghitung proses perubahan bobot. Dari hasil simulasi komputer maka diperoleh perbandingan antara algoritma backpropagation standar dengan backpropagation dengan penambahan momentum. Untuk algoritma backpropagation standar kecepatan konvergensi 727 epoch dengan nilai MSE 0,01, sedangkan algoritma backpropagation standar mencapai 4000 epoch dengan nilai MSE 0,001. Hal ini menunjukkan bahwa algoritma backpropagation adaptive learning lebih cepat mencapai konvergensi daripada algoritma backpropagation standar.

Keywords— Backpropagation, Adaptive learning rate, momentum, konvergensi.

I. PENDAHULUAN

Perkembangan teknologi ilmu komputer saat ini telah menciptakan beberapa teknik pendekatan dalam menyelesaikan suatu masalah yang disebut soft computing. Soft Computing merupakan bagian dari sistem cerdas yang merupakan suatu model pendekatan untuk melakukan komputasi dengan meniru akal manusia dan memiliki kemampuan untuk menalar dan belajar pada lingkungan yang penuh dengan ketidakpastian dan ketidaktepatan. Komponen utama pembentuk soft computing adalah sistem fuzzy (fuzzy system), jaringan syaraf (neural network), algoritma evolusioner (evolutionary algorithm), dan penalaran dengan probabilitas (probabilistic reasoning).

Multi layer perceptron terdiri dari input layer, hidden layer dan output layer. masing-masing layer terdiri dari node. Banyak model pembelajaran pada menggunakan algoritma backpropagation. Algoritma ini juga berhasil diterapkan pada multi layer perceptron untuk menyelesaikan persoalan yang luas. Metode backpropagation standar (Gradient Descent) punya keterbatasan yaitu laju konvergensi yang cukup lambat[1]. Laju konvergensi dari algoritma backpropagation juga tergantung pada pemilihan arsitektur jaringan, bobot awal dan bias, learning rate, koefisien momentum dan fungsi aktivasi[2]. Oleh karena itu dalam penelitian ini modifikasi dilakukan terhadap backpropagation standar dengan cara mengganti fungsi pelatihannya. Modifikasi yang akan digunakan adalah pemilihan arsitektur yang tepat serta metode yang menggunakan adaptive laju pembelajaran dan momentum. Algoritma ini memiliki performa pelatihan yang tinggi sehingga mempercepat laju

konvergensi. Dalam hal perkebunan juga prediksi dilakukan dengan pembelajaran backpropagasi adapun peneitian yakni memorisasi mampu mengenali 90.74% dan generalisasi mampu mengenali sebesar 84.44%, bidang perkebunan [3] memprediksi produktivitas lahan perkebunan kelapa sawit berdasarkan beberapa anasir tanah dan iklim secara simultan, curah hujan menggunakan metode backpropagation untuk memprediksi cuaca, dan menemukan bahwa proses pelatihan dapat dilakukan dengan cepat. Algoritma Backpropagation merupakan algoritma yang memiliki komponen utama yang akan melakukan proses pembelajaran dalam melakukan sebuah prediksi. Semakin besar nilai learning rate mengakibatkan semakin besarnya langkah pembelajaran. Jika learning rate diset terlalu besar, maka algoritma akan menjadi tidak stabil, dan sebaliknya algoritma akan konvergen dalam jangka waktu yang sangat lama. Untuk meningkatkan kecepatan proses training dilakukan dengan menambahkan parameter momentum. Range nilai yang umum digunakan antara 0 sampai dengan 1. Nilai yang biasanya paling optimal adalah 0,9. Momentum ini berguna untuk mempercepat konvergensi.

Penggunaan algoritma pelatihan dalam jaringan syaraf tiruan telah dilakukan dalam menyelesaikan banyak permasalahan prediksi produktivitas kelapa sawit. Peramalan produksi kelapa sawit terbaik diperoleh dengan kombinasi umur tanaman, pemupukan, jumlah hari hujan dan penyinaran matahari pada 18 BSP, dan defisit air pada 24 BSP. Regresi terbaik adalah $Y = 3.15 + 0.010 \text{ umur tanaman} - 0.016 \text{ pupuk} - 0.016 \text{ penyinaran} - 0.005 \text{ defisit air} -$

0.015 hari hujan. Faktor agroekologi menentukan akurasi ramalan produksi kelapa sawit.

Berdasarkan kajian permasalahan yang telah disampaikan, maka tujuan penelitian ini adalah menerapkan algoritma pelatihan backpropagation dalam contoh kasus prediksi produktivitas kelapa sawit untuk mendapatkan tingkat error, menentukan pelatihan jaringan yang paling optimal (yang memberikan error terkecil) dan kecepatan konvergensi jika momentum tersebut dinaikkan atau diturunkan nilainya. Dari uraian diatas, penulis tertarik untuk mengambil judul “Analisis Penambahan Nilai Momentum Pada Prediksi Produktivitas Kelapa Sawit Dengan Backpropagasi Neural Network”

II. LANDASAN TEORI

A. Neural Network

Artificial Neural Network / Jaringan Saraf Tiruan (JST) adalah paradigma pengolahan informasi yang terinspirasi oleh sistem saraf secara biologis, seperti proses informasi pada otak manusia. Elemen kunci dari paradigma ini adalah struktur dari sistem pengolahan informasi yang terdiri dari sejumlah besar elemen pemrosesan yang saling berhubungan (*neuron*), bekerja serentak untuk menyelesaikan masalah tertentu.

Neuron biologi merupakan sistem yang “*fault tolerant*” dalam 2 hal. Pertama, manusia dapat mengenali sinyal input yang agak berbeda dari yang pernah kita terima sebelumnya. Sebagai contoh, manusia sering dapat mengenali seseorang yang wajahnya pernah dilihat dari foto, atau dapat mengenali seseorang yang wajahnya agak berbeda karena sudah lama tidak dijumpainya.

Kedua, otak manusia tetap mampu bekerja meskipun beberapa neuronnya tidak mampu bekerja dengan baik. Jika sebuah *neuron* rusak, *neuron* lain kadang-kadang dapat dilatih untuk menggantikan fungsi sel yang rusak tersebut.

Cara kerja JST seperti cara kerja manusia, yaitu belajar melalui contoh. Lapisan-lapisan penyusun JST dibagi menjadi 3, yaitu lapisan input (*input layer*), lapisan tersembunyi (*hidden layer*), dan lapisan output (*output layer*). Jaringan syaraf tiruan adalah merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba untuk mensimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan disini digunakan karena jaringan syaraf ini diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran.

B. Metode Backpropagation

d Backpropagation merupakan salah satu dari beberapa metode yang digunakan dalam JST dan yang paling sering digunakan dalam berbagai bidang aplikasi, seperti pengenalan pola, peramalan dan optimisasi. Jaringan syaraf tiruan *backpropagation* (BP) pertama kali diperkenalkan oleh Rumelhart, Hinton dan William pada tahun 1986, kemudian

Rumelhart dan Mc Clelland mengembangkannya pada tahun 1988. Jaringan ini merupakan model jaringan multilayer. *backpropagation* paling banyak digunakan oleh pemakai jaringan saraf tiruan, bahkan diperkirakan lebih dari 80 % proyek jaringan saraf tiruan yang tengah dikembangkan menggunakan *backpropagation* [4].

Arsitektur *backpropagation* merupakan salah satu arsitektur jaringan saraf tiruan yang dapat digunakan untuk mempelajari dan menganalisis pola data masa lalu lebih tepat sehingga diperoleh keluaran yang lebih akurat (dengan kesalahan atau *error* minimum).

Langkah-langkah dalam membangun algoritma *backpropagation* adalah sebagai berikut:

- a. Inisialisasi bobot (ambil nilai *random* yang cukup kecil).
- b. Tahap perambatan maju (*forward propagation*)
 - 1) Setiap unit *input* ($X_i, i=1,2,3,...,n$) menerima sinyal x_i dan meneruskan sinyal tersebut ke semua unit pada lapisan tersembunyi.
 - 2) Setiap unit tersembunyi ($Z_j, j=1,2,3,...,p$) menjumlahkan bobot sinyal *input*, ditunjukkan dengan persamaan (1).

$$z_{in_j} = v_{0j} + \sum_{i=1}^n x_i v_{ij} \quad (1)$$
 Dan menerapkan fungsi aktivasi untuk menghitung sinyal *output*-nya, ditunjukkan dengan persamaan (2).

$$z_j = f(z_{in_j}) \quad (2)$$
 Fungsi aktivasi yang digunakan adalah fungsi *sigmoid*, kemudian mengirimkan sinyal tersebut ke semua unit *output*.
 - 3) Setiap unit *output* ($Y_k, k=1,2,3,...,m$) menjumlahkan bobot sinyal *input*, ditunjukkan dengan persamaan (3).

$$y_{in_k} = w_{0k} + \sum_{j=1}^p z_j w_{jk} \quad (3)$$
 Dan menerapkan fungsi aktivasi untuk menghitung sinyal *output*-nya, ditunjukkan dengan persamaan (4).

$$y_k = f(y_{in_k}) \quad (4)$$
- c. Tahap perambatan balik (*backpropagation*)
 - 1) Setiap unit *output* ($Y_k, k=1,2,3,...,m$) menerima pola target yang sesuai dengan pola *input* pelatihan, kemudian hitung *error*, ditunjukkan dengan persamaan (5).

$$\delta_k = (t_k - y_k) f'(y_{in_k}) \quad (5)$$
 f' adalah turunan dari fungsi aktivasi. Kemudian hitung korelasi bobot, ditunjukkan dengan persamaan (6).

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (6)$$
 Dan menghitung koreksi bias, ditunjukkan dengan persamaan (7).

$$\Delta w_{0k} = \alpha \delta_k \quad (7)$$
 Sekaligus mengirimkan δ_k ke unit-unit yang ada di lapisan paling kanan.
 - 2) Setiap unit tersembunyi ($Z_j, j=1,2,3,...,p$) menjumlahkan delta *input*-nya (dari unit-unit yang berada pada lapisan di kanannya), ditunjukkan dengan persamaan (8).

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk} \quad (8)$$

Untuk menghitung informasi *error*, kalikan nilai ini dengan turunan dari fungsi aktivasinya, ditunjukkan dengan persamaan (9).

$$\delta_j = \delta_{in_j} f'(z_{in_j}) \quad (9)$$

Kemudian hitung koreksi bobot, ditunjukkan dengan persamaan (10).

$$\Delta v_{jk} = \alpha \delta_j x_k \quad (10)$$

Setelah itu, hitung juga koreksi bias, ditunjukkan dengan persamaan (11).

$$\Delta v_{0j} = \alpha \delta_j \quad (11)$$

d. Tahap perubahan bobot dan bias

- 1) Setiap unit output ($Y_k, k=1,2,3,\dots,m$) dilakukan perubahan bobot dan bias ($j=0,1,2,\dots,p$), ditunjukkan dengan persamaan (12).

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \quad (12)$$

Setiap unit tersembunyi ($Z_j, j=1,2,3,\dots,p$) dilakukan perubahan bobot dan bias ($i=0,1,2,\dots,n$), ditunjukkan dengan persamaan (13).

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \quad (13)$$

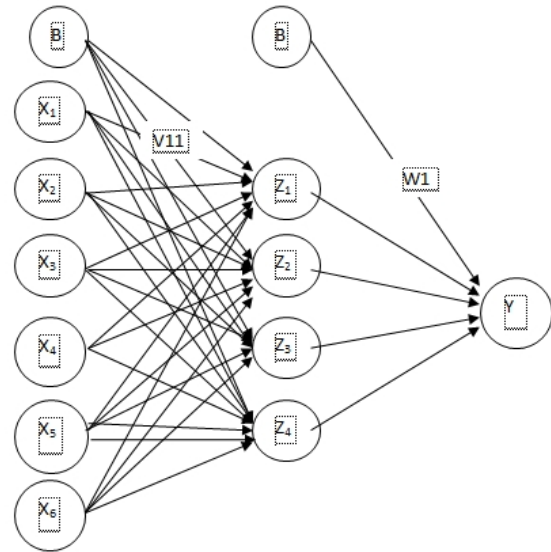
- 2) Tes kondisi berhenti.

III. METODE PENELITIAN

penelitian ini akan dilakukan beberapa langkah untuk mencapai tujuan penelitian. Langkah – langkah tersebut dapat digambarkan melalui diagram alir pada gambar 1 Penelitian ini dimulai dari pengumpulan data dari data mentah. Tahap berikutnya adalah penentuan desain arsitektur jaringan dengan penentuan pola masukan dan keluaran untuk keperluan pelatihan dan pengujian pada jaringan syaraf tiruan (JST) Tahap ini kemudian diikuti dengan penentuan algoritma pelatihan.

Selanjutnya adalah tahap pelatihan terhadap data yang telah dinormalisasi dan ditentukan arsitekturnya, pelatihan dilakukan pertama untuk algoritma *backpropagation* standar, setelah itu baru dilakukan kembali pelatihan dengan menambahkan *learning rate* dan *koefisien momentum* pada update bobot. Tujuan pelatihan tersebut untuk membandingkan nilai *epoch* dan penentuan nilai *Mean Square Error (MSE)*. Setelah dilakukan tahap pelatihan adalah tahap pengujian terhadap data pengujian, tujuannya untuk mengetahui tingkat validasi hasil.

Desain arsitektur jaringan dilakukan untuk prediksi produktivitas kelapa sawit dimulai dengan menentukan banyaknya data masukan yang digunakan, banyaknya layer tersembunyi (*hidden layer*) yang digunakan, dan banyaknya keluaran yang diinginkan. Data yang digunakan sebagai masukan sebanyak 6 data input dan data keluaran atau target adalah data pada tahun ke-4 (data input 2010 – 2014 dengan target 2015). Untuk mengetahui umur, pemupukan, kelembaban, penyinaran, curah hujan, dan hari hujan pada tahun ke-5 maka data masukannya merupakan data pada tahun ke-4 sampai tahun ke-5 (data input 2014 – 2015 dengan target 2016), demikian seterusnya.



Gbr.1 Desain *Backpropagation Neural Network*

A. Pengenalan pola (pelatihan)

Pengenalan pola dilakukan dengan cara penyesuaian nilai bobot. Penghentian penyesuaian bobot dalam pengenalan pola apabila *error* yang dihasilkan mencapai *target error*. *Error* dihitung setelah tahapan *forward propagation*. Apabila *error* lebih besar dari *target error* maka pelatihan akan dilanjutkan ke tahap *backward propagation* sampai *error* yang dihasilkan mencapai *target error*.

B. Pengujian dan prediksi

Jaringan yang telah dilatih dan mencapai hasil yang dikehendaki perlu diuji untuk mengetahui kemampuannya pada saat mempelajari data latih yang diberikan. Pengujian dapat dilakukan dengan menggunakan data set yang sudah dilatihkan untuk melihat unjuk kerja sistem aplikasi yang telah dibuat dengan melihat nilai *error* minimumnya. Selain itu juga pengujian dapat dilakukan menggunakan data set yang belum pernah dilatihkan sebelumnya untuk melihat tingkat akurasi sistem yang telah dibuat, yaitu menggunakan data uji mulai data tahun 2014. Pada Proses validasi, sistem diuji dengan data lain, hal ini dimaksud untuk mengetahui sejauh mana sistem dapat menginformasikan nilai – nilai keluaran dari nilai – nilai masukan yang diberikan.

IV. HASIL DAN PEMBAHASAN

A. Pengujian dan prediksi

Untuk melakukan pembobotan harus memasukan data terlebih dahulu. Dalam inisialisasi pembobotan awal dapat dilakukan dengan metode *Backpropagation*. Dalam pemberhentian program makan dibutuhkan 2 cara yaitu menggunakan *epoch* dan *error* untuk mendapatkan nilai yang dicapai.

Dalam melakukan pengisian data maka dilakukan pembobotan yang konstanta. Pada saat tahap pembobotan akan dihitung bobot dan bias untuk melakukan pelatihan. Tahap pembobotan ini akan

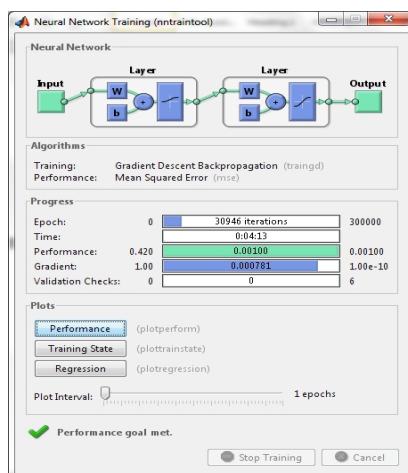
dipilih secara random, yang mana bobot yang diperoleh akan digunakan untuk *feedforward* (arus maju). Dimana pada tahap ini akan menerima sinyal X (data yang mempengaruhi penambahan nilai momentum pada prediksi produktivitas kelapa) Sinyal yang dimasukan akan diterima dan dikalikan dengan bobot pada satu *node* dari *input layer* menuju *hidden layer* serta ditambah dengan bias. Setelah tahap ini dilakukan pada masing-masing *node* pada *hidden* akan dihasilkan sinyal bobot pada satu *node hidden layer*. Untuk menghitung sinyal *output* pada *hidden layer* digunakan fungsi aktivasi *sigmoid* dan *threshold* untuk *hidden layer*.

Setelah itu akan menjumlahkan bobot dari sinyal *input* sehingga didapat sinyal *output* dari *output layer* yang sudah diaktifkan. Sinyal yang diperoleh dari *output layer* akan dihitung *error*-nya dengan mengurangkan dengan data target. Selisih pengurangannya disebut dengan nilai *error*. Nilai *error* harus dicari nilainya lebih kecil dari batas *error* yang digunakan. Jika nilainya masih diatas batas *error* maka dilakukan koreksi bobot dan bias, koreksi bobot dan bias dilakukan untuk mengurangi nilai *error* sehingga sistem menemukan pola untuk mendapatkan target. Selanjutnya bobot yang dapat menemukan pola untuk prediksi akan disimpan.

Pada tahap pengujian bobot yang diperoleh pada saat pembobotan akan digunakan untuk menguji sistem, apakah sistem sudah dapat menemukan target atau belum. Dalam pengujian ini dilakukan sampai diperoleh *error* paling rendah atau yang mendekati target yang dicapai.

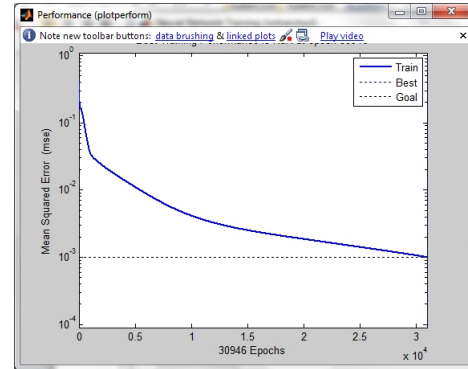
B. Training Data Algoritma Backpropagation Standar

Setelah melakukan pembobotan serta penginputan data maka akan dilakukan pelatihan data. Dalam proses pelatihan akan dilihat dari jumlah *Epoch* dan batas *error* yang didapat. Untuk tahap ini menggunakan training menggunakan arsitektur 6-4-1 untuk medapatkan pola tersebut. Dapat lihat dari gambar 2 berikut ini :



Gbr.2 Pelatihan menggunakan arsitektur 6-4-1

Pada pelatihan ini untuk mendapatkan target menggunakan waktu 4 menit 3 detik dengan *epoch* 30946 *iterations* dimana performance (*goal*) yang digunakan 0,0001. Sedangkan dalam bentuk grafik yang terlihat pada gambar berikut :



Gbr. 3 Grafik pelatihan menggunakan arsitektur 6-4-1

Dari hasil training ini dari 6 variabel yang didapat maka akan mendapatkan nilai *output* dari target yang ditentukan serta MSE (*Mean Square Error*) seperti terlihat pada tabel I berikut ini :

TABEL I
OUTPUT, ERROR DAN MSE

NO	Nama	Target	output	Error	Mse
1	Januari	0,1	0,1082	-0,0082	0,0000672400
2	Februari	0,1	0,1198	-0,0198	0,0003920400
3	Maret	0,4	0,4038	-0,0038	0,0000144400
4	April	0,2	0,1712	0,0288	0,0008294400
5	Mei	0,3	0,303	-0,003	0,0000090000
6	Juni	0,6	0,5346	0,0654	0,0042771600
7	Juli	0,6	0,6164	-0,0164	0,0002689600
8	Agustus	0,6	0,6457	-0,0457	0,0020884900
9	September	0,8	0,7942	0,0058	0,0000336400
10	Oktober	0,8	0,8224	-0,0224	0,0005017600
11	Nopember	0,7	0,7271	-0,0271	0,0007344100
12	Desember	0,9	0,8472	0,0528	0,0027878400

Nilai *output* langsung diperhitungkan dengan menggunakan matlab, dengan *source code* :
[a,Pf,Af,e,Perf]=sim(net,p,[],t)
apabila dengan perhitungan menggunakan rumus :

$$y_{ink} = b2k + \sum_{i=1}^M x_i v_{jk}$$

Sedangkan untuk mendapatkan nilai error dari target menggunakan rumus :

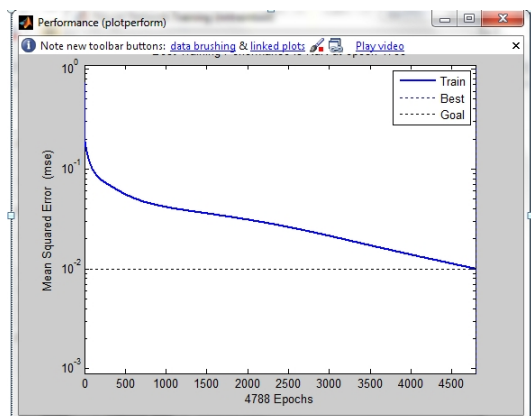
$$E = t - y_k$$

Untuk mendapkatka hasil dari MSE (*Mean Square Error*) menggunakan rumus :

$$MSE = \frac{\sum E^2}{n}$$

C. Hasil Pelatihan Algoritma Backpropagation standar

Pada pelatihan algoritma backpropagation data set yang digunakan Terdapat 6 input data yaitu data umur, pemupukan, kelembaban, curah hujan dan hari hujan dalam memprediksi produktivitas kelapa sawit. Arsitektur yang dipilih adalah 6-4-1 dengan target error ditetapkan 0,01. Hasil pelatihan yang diperoleh pada saat pembelajaran jaringan mencapai konvergensi pada epoch ke 4728, dengan nilai mean square error yang dihasilkan sebagai indikator kinerja jaringan syaraf mencapai 0,001 pembelajaran di perhatikan pada gambar 4 Gambar Hasil Pelatihan Algoritma Pada pelatihan algoritma backpropagation digunakan memprediksi produktivitas kelapa sawit 0,01. Pada pembelajaran ini dengan metode yang digunakan dengan menavariabel learning rate nilai learning rate Grafik pembelajaran backpropagation Backpropagation learning rate yang terdiri dari Arsitekturyang dipilih adalah6-4-1dengan target dan koefisien momentum pada fase perubahan bobot, dimana di melakukan penyesuaian secara adaptif pada proses perubahan. Grafik hasil

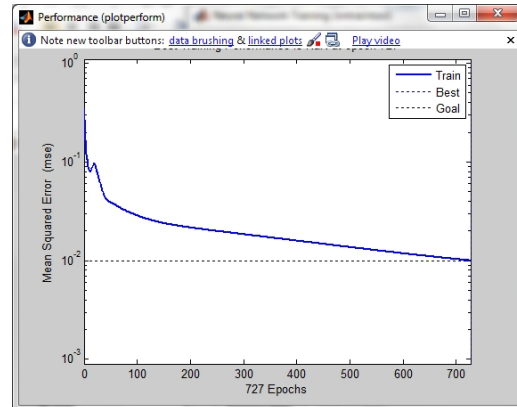


Gbr. 4 Grafik pembelajaran backpropagasi standar

D. Hasil Pelatihan Algoritma Backpropagation dengan momentum

Pada pelatihan algoritma backpropagation data set yang digunakan Terdapat 6 input data yaitu data umur, pemupukan, kelembaban, curah hujan dan hari hujan dalam memprediksi produktivitas kelapa sawit. Arsitektur yang dipilih adalah 6-4-1 dengan target error ditetapkan 0,01 dengan penambahan momentum. Hasil pelatihan yang diperoleh pada saat pembelajaran jaringan mencapai konvergensi pada epoch ke 727, dengan nilai mean square error yang dihasilkan sebagai indikator kinerja jaringan syaraf mencapai 0,001 pembelajaran di perhatikan pada gambar 5. Gambar Hasil Pelatihan Algoritma Pada pelatihan algoritma backpropagation digunakan memprediksi produktivitas kelapa sawit 0,01. Pada pembelajaran

ini dengan metode yang digunakan dengan menavariabel learning rate nilai learning rate Grafik pembelajaran backpropagation Backpropagation learning rate yang terdiri dari Arsitekturyang dipilih adalah6-4-1dengan target dan koefisien momentum pada fase perubahan bobot, dimana di melakukan penyesuaian secara adaptif pada proses perubahan. Grafik hasil



Gbr. 5 Grafik pembelajaran backpropagasi momentum

E. Hasil Pengujian

Setelah proses pelatihan dilakukan pada algoritma backpropagation adaptive learning pengujian data untuk algoritma diperoleh tingkat akurasi sebesar 86%.

F. Pembahasan

Berdasarkan grafik yang dihasilkan dari pelatihan data terlihat bahwa laju konvergensi jaringan algoritma backpropagation konvergensi pada algoritma menambahkan parameter pada percepatan pembelajaran backpropagation. Gambar 5 backpropagation maka tahap berikutnya pengujian data, hasil backpropagation standar dan standar sangat lambat dibandingkan dengan laju backpropagation adaptive learning. Ini menunjukkan bahwa adaptive learning dan koefisien momentum sangat berpengaruh (accelerated learning) pada jaringan syaraf tiruan Grafik perbandingan kedua algoritma standar dan adaptive learning Pada grafik 5 diperlihatkan perbedaan iterasi yang dihasilkan dari kedua algoritma dengan nilai epoch yang jauh berbeda. Algoritma yang diusulkan (adaptive learning rate) hanya membutuhkan 727 epoch serta hasil MSE yang diperoleh mendekati 0 untuk mencapai titik konvergensi menunjukkan jauh lebih cepat dibandingkan dengan algoritma backpropagation standar yang mencapai 1000 epoch, menunjukkan bahwa kinerja algoritma backpropagation adaptive learning lebih baik dibandingkan algoritma backpropagation standar. Percepatan pembelajaran pada algoritma backpropagation adaptive learning disebabkan penambahan learning rate secara adaptif disesuaikan dengan perubahan perhitungan kuadrat error dan koefisien momentum pada setiap iterasi sehingga lebih cepat mencapai error.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Dari hasil penelitian dan pembahasan maka penulis mengambil kesimpulan sebagai berikut.

1. Penerapan algoritma jaringan syaraf tiruan *backpropagation* secara luastelah di implementasikan dalam berbagai aplikasi yang praktis, algoritma ini masih memerlukan perbaikan. Pada penelitian ini penulis menambahkan parameter *adaptive learning rate* dan koefisien momentum pada perhitungan update bobot dengan mengambil nilai bobot pada iterasi yang lalu dan menambahkan nilai bobot sekarang untuk meningkatkan percepatan pembelajaran (*accelerated learning*) pada algoritma *backpropagation*.
2. Efektifitas algoritma dengan membandingkan hasil iterasi antara algoritma *backpropagation standar* dengan *backpropagation adaptive learning*
3. Dari hasil penelitian menunjukkan bahwa algoritma *backpropagation* dengan *penambahan nilai momentum* hanya mencapai 727 *epoch* dengan nilai MSE 0,01, sedangkan algoritma *backpropagation standar* mencapai 4000 *epoch* dengan nilai MSE 0,001.

B. Saran

Berdasarkan hasil penelitian maka saran yang dapat disampaikan adalah sebagai berikut :

1. Untuk meningkatkan percepatan pembelajaran maka penentuan bobot awal untuk pelatihan dapat menggabungkan dengan algoritma yang lain untuk keakuratan penentuan bobot awal.
2. Modifikasi pada pelatihan standar *backpropagation* selain menggunakan *adaptive learning* dan *penambahan nilai momentum* perlu dicoba untuk mempercepat iterasi selama proses pelatihan jaringan.

REFERENCE

- [1] Hamid N.A&Nawi, N.M(2011).*The Effect of Gain Variation of Activation Function in Improving Training Time of Back Propagation Neural Network on Classification Problems*. Proceedings of the KolokiumKebangsaaan Pasca SiswazahSain dan Matematik 2009.UPSI:pp.
- [2] Hamid N.A, Nawi, N.M, Ghazali, R. & Saleh, M N.M. (2011) *Accelerating Learning Performance of Backpropagation Algorithm by Using Adaptive Gain Together with Adaptive Momentum and Adaptive Learning Rate on Classification Problems*. International Journal of Software Engineering and Application.Vol 5 No. 4.
- [3] Hermantoro Dkk 2009, Aplikasi Model Artificial Neural Network (Ann) Terintegrasi Dengan Geographycal Information System (Gis) Untuk Evaluasi Kesesuaian Lahan Perkebunan Kakao1.
- [4] Muhammad Erwin Ashari Haryono, "Model Identifikasi Peta Secara Otomatis Menggunakan Konsep Jaringan Syaraf Tiruan Backpropagation" Viewed 1 Juni2004.